

PLC-styrning av Inteco Multitank

PLC-styrning med HMI-interface för flertankssystem

Markus Enqvist

Examensarbete för automationsingenjör (YH)-examen

Utbildningsprogrammet för el- och automationsteknik

Vasa 2022

EXAMENSARBETE

Författare: Markus Enqvist

Utbildning och ort: El- och automationsteknik, Vasa

Inriktning: Automationsteknik

Handledare: Joachim Böling

Titel: PLC-styrning av Inteco Multitank

Datum: 27.1.2022 Sidantal: 17

Bilagor: 1

Abstrakt

Examensarbetets mål var att skapa ett PLC-program med HMI-interface som kunde ersätta ett befintligt MATLAB-program som används för att styra ett Inteco flertankssystem. Programmeringen gjordes i Siemens TIA-portal och överfördes till en Siemens S7 1215C-kontroller som anslöts till en SIMATIC KTP700 Basic HMI-skärm. Programmet skulle ha liknande funktioner som det MATLAB-program som fanns tillgängligt från tidigare.

Resultatet är ett PLC-program som styrs via en HMI-skärm, ett användargränssnitt till HMI-skärmen samt en bruksanvisning för framtida användare. Några av funktionerna i programmet är manuell styrning av enskilda komponenter, visuell överblick av vattennivåerna i tanken, PWM-kontroll, kalibrering av enskilda komponenter samt PID-styrning.

Språk: svenska

Nyckelord: plc, multitank, pid

BACHELOR'S THESIS

Author: Markus Enqvist

Degree Programme: Electricity and Automation

Specialisation: Automation

Supervisor(s): Joachim Böling

Title: PLC-control of Inteco Multitank

Date: 27.1.2022 Number of pages: 17

Appendices: 1

Abstract

The goal of the thesis was to create a PLC-program with an HMI interface that could replace an existing MATLAB program which is used to control an Inteco multitank system. The programming was done in Siemens TIA portal and was transferred to a Siemens S7 1215C controller which was connected to a SIMATIC KTP700 Basic HMI screen. The program should have similar functions to the already existing MATLAB program.

The result is a PLC-program controlled from an HMI-screen, an interface to the HMI-screen and a manual for future users. Some of the functions in the program are manual control of separate components, visual overview of the water levels in the tank, PWM control, calibration of separate components and PID control.

Language: Swedish

Key words: plc, multitank, pid

Förkortningar

AI analog input

HMI människa-maskingränssnitt

PC persondator

PID proportionell, integrerande, deriverande

PLC programmerbart styrsystem

PWM pulsbreddsmodulering

Innehållsförteckning

1	Inledning.....	1
2	Problem.....	1
3	Teori.....	3
3.1	PLC.....	3
3.2	PWM.....	4
3.3	PID.....	5
3.4	HMI.....	7
3.5	Liknande system.....	8
4	Metoder.....	8
5	Resultat.....	16
6	Diskussion.....	17
7	Sammanfattning.....	17
7.1	Skribentens egna anmärkingar.....	18
8	Källor.....	18

1 Inledning

För att säkra användningen i framtiden av det Inteco flertankssystem (Figur 1) som finns tillgängligt i Technobothnia ville Yrkeshögskolan Novia att det existerande MATLAB-programmet som kördes på en äldre PC skulle ersättas av en PLC med HMI-interface. Projektet är en fortsättning på ett tidigare praktikprojekt där interface-elektroniken som förstärkte signalerna från nivågivarna i flertankssystemet tillverkades. Målet var att skapa ett intuitivt gränssnitt där man enkelt kan styra de viktigaste funktionerna för systemet.

Eftersom programmet skapades i TIA-portal kan man enkelt lägga till fler funktioner i ett framtida projekt för att ytterligare utöka användbarheten av flertankssystemet.

I examensarbetet berättas först om några av de funktioner som programmet har och teorin bakom dessa funktioner, sedan reflekteras det på hur programmet kan utökas i framtiden.

Som bilaga bifogas en användarhandbok som skrevs för att underlätta användningen av HMI-gränssnittet för slutanvändare.

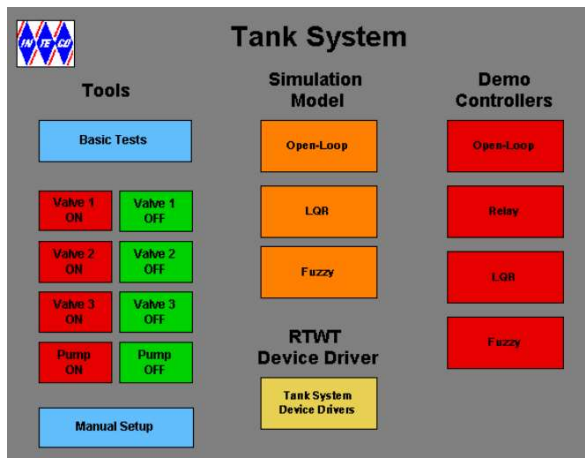
2 Problem

Flertankssystemet som fanns tillgängligt på Technobothnia kördes via en äldre stationär dator med en föråldrad MATLAB-version som inte längre går att köpa. Eftersom programmet inte fungerade på den moderna versionen av MATLAB ville Yrkeshögskolan Novia skapa ett liknande program för en PLC så att flertankssystemet kunde användas utan att förlita sig på den stationära datorn.



Figur 1. Flertankssystem.

Eftersom det existerande programmet kördes i MATLAB fanns det ingen tillgänglig programkod som direkt kunde överföras till en PLC-miljö, utan alla funktioner skulle återskapas från noll. Med hjälp av bruksanvisningen för flertankssystemet [1] och funktioner som fanns tillgängliga i Siemens TIA-portal skulle ett helt nytt program skapas som kunde utföra liknande funktioner som MATLAB-programmet (Figur 2).



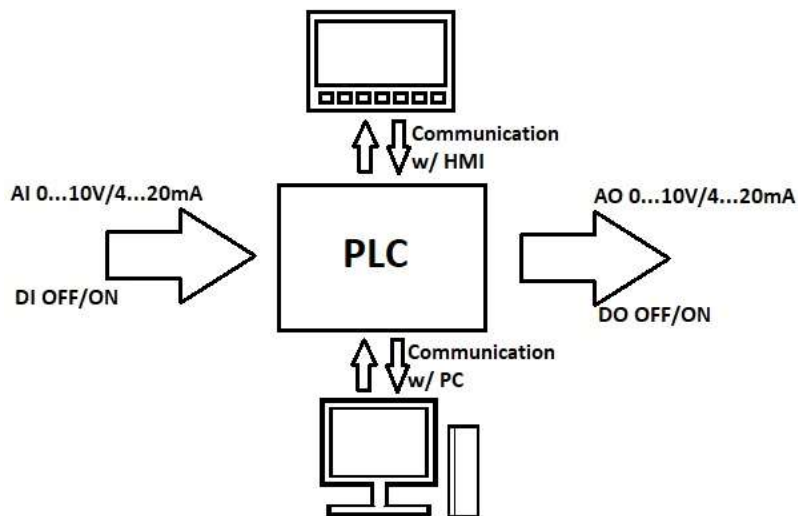
Figur 2. MATLAB-program för flertankssystemet.

3 Teori

Här förklaras mer ingående vad PLC, PWM, PID och HMI egentligen är för att ge läsaren en bättre inblick i projektet samt dess olika funktioner. Dessa begrepp är väsentliga för projektet och viktiga för läsaren att bekanta sig med.

3.1 PLC

En PLC är en controller som används överallt i industrimiljö för att bland annat övervaka processer och/eller styra automatik. En PLC kan ha digitala- och/eller analoga ingångar och utgångar (Figur 3). Skillnaden mellan dessa är att analoga signaler kan ha varierande värden, 0...10 V / 0...20 mA, medan digitala ingångar är på eller av, 0/24 V. [3]



Figur 3. Flödesdiagram för PLC.

En PLC används för att utföra specifik programkod, vilket gör dem mer energieffektiva och pålitliga än till exempel datorstyrda system. En PLC har inget operativsystem som drar systemresurser utan den utför endast de uppgifter som den programmeras till.

En annan fördel med PLC-system är att de lämpar sig bra för de tuffa industrimiljöerna var övriga styrsystem skulle vara oanvändbara. En PLC har inga rörliga komponenter som fläktar och hårddiskar vilket gör dem bättre lämpade i tuffa miljöer med sämre ventilation, starka vibrationer och övriga yttre påfrestningar [4].

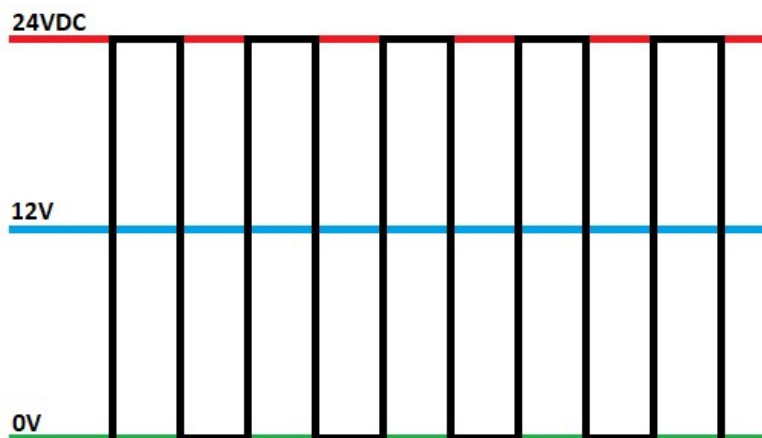
Den PLC som användes i detta projekt var en Siemens S7 1200, modell 1215C. Denna PLC har 14 digitala ingångar, 10 digitala utgångar, 2 analoga ingångar och 2 analoga utgångar [2].

I projektet användes 3 analoga ingångar, en för varje nivågivare i flertankssystemet, därför behövdes ytterligare en analog ingång utifrån de som fanns tillgängliga på vår PLC. Här användes en SM1231 AI tilläggsmodul för att få tillräckligt många analoga ingångar.

I projektet användes 4 digitala utgångar för att styra ventilerna och pumpen i flertankssystemet. För att kunna reglera dessa komponenter med digitala signaler användes pulsbreddsmodulering.

3.2 PWM

PWM är en metod man kan använda för att kunna reglera komponenter med en digital signal. I stället för att en signal är konstant på eller av kan man med pulsbreddsmodulering reglera signalen genom pulsering. Med hjälp av pulsbreddsmodulering kan man reglera komponenter genom att skicka snabba av och på pulser som i sin tur minskar effekten hos komponenten utan analoga signaler, till exempel om du har en 24 V LED-lampa som lyser för starkt kan du med hjälp av pulsbreddsmodulering med 50 % tillslagstid få lampan att lysa hälften så starkt och förbruka hälften av effekten (Figur 4). Om detta filmas med en höghastighetskamera kan man se att lampan blinkar, men detta sker så snabbt att det mänskliga ögat uppfattar det som att lampan lyser svagare. [5].



Figur 4. 24 V PWM-signal med 50 % tillslagstid.

Pulsbreddsmodulering kan bland annat användas för att dimma LED-belysning, justera hastigheten av en DC-elmotor eller i detta fall justera flödet genom ventiler.

För att en pulsbreddsmodulering ska fungera korrekt måste man ange en förutbestämd cykeltid utifrån komponenterna som ska styras. Därefter regleras komponenterna genom en variabel tillslagstid, vilket innebär hur stor andel av cykeltiden som pulserna ska vara höga. Ifall en pulsbreddsmodulering har en 1000 ms cykeltid och man vill att en komponent ska köras på 50 % skulle komponenten under hälften av cykeltiden, 500 ms, vara påslagen och under resten av tiden vara avslagen.

3.3 PID

PID-reglering är en regulatorkonstruktion som använder en matematisk formel för att nå ett önskat börvärde genom att samla sensorinformation med ett jämnt tidsintervall och med hjälp av de ändrade värdena justera reglaget för att nå önskad nivå. Beroende på hur P-, I- och D-parametrarna väljs kan man justera beteendet hos regulatorn.

En PID-regulator kan vara antingen kontinuerlig, direkt sampling av analoga signaler i nutid, eller diskret, digital sampling av signaler med en viss samplingstid.

För att skapa en diskret PID-regulator måste man transformera signalerna med hjälp av Eulers stegmetod eller bilinjär transformation.

Formeln för en kontinuerlig PID-reglering är följande:

$$u(t) = Kp * (r(t) - y(t) + \frac{1}{Ti} * \int_0^t r(t) - y(t) * dt + Td * (\frac{d}{dt} * (r(t) - y(t)))) \quad (1)$$

$u(t)$ =styrsignalen från PLC vid tiden t , Kp =proportionella förstärkningen, Ti =integrationstiden, Td =deriveringstiden, $r(t)$ =börvärde, $y(t)$ =ärvärde, dt =samplingstid.

Formeln för en diskret (digital) PID-regulator transformerad med Eulers stegmetod:

$$u(t) = Kp * (r(t) - y(t) + (\frac{1}{Ti} * dt) + Td * (\frac{z-1}{dt} * (r(t) - y(t)))) \quad (2)$$

Några av utmaningarna med diskretisering är att få en så nära uppskattning av det analoga värdet som möjligt samt att försöka få regulatorn så snabb som möjligt.

Beroende på vilken typ av transformationsformel man använder sig av kan man behöva tillägga en kort fördröjning för att förhindra räknefel, till exempel att formeln försöker sampla ett tidigare mätvärde som ej existerar.

Med dagens elektronik kan man sampla mätvärden så otroligt snabbt att skillnaderna mellan kontinuerliga PID-regulatorer och diskreta PID-regulatorer är knappt märkbara.

En PID-regulator försöker nå börvärdet, det vill säga den önskade nivån, genom att ta differensen av börvärdet och ärvärdet och utifrån parametrarna P, I och D styra processen mot börvärdet (Figur 5).

P, proportionell förstärkning, multipliceras med differensen av nuvarande ärvärdet och börvärdet.

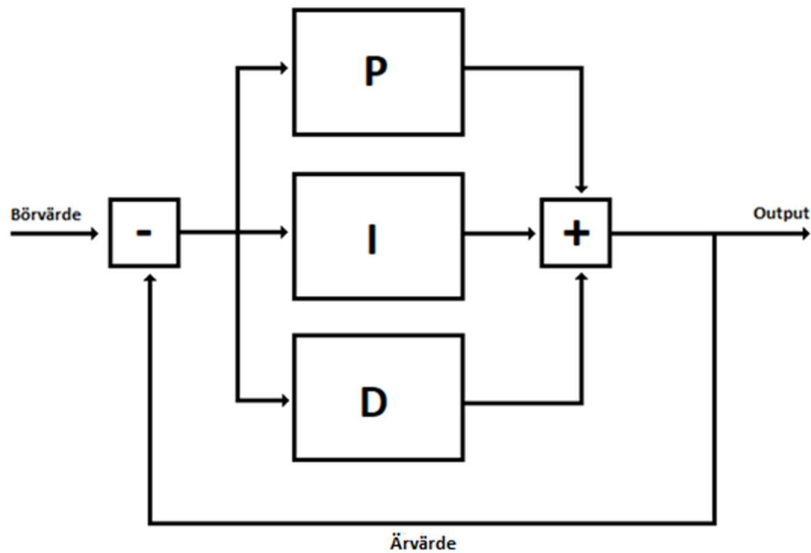
$$Kp * (r(t) - y(t)) \quad (3)$$

I, integrerande förstärkning, multiplicerar summan av skillnaden mellan tidigare ärvärden och börvärden multiplicerat med samplingstiden med inversen av integrationstiden.

$$\frac{1}{Ti} * \int_0^t r(t) - y(t) * dt \quad (4)$$

D, deriverande förstärkning, multipliceras med derivatan av samplingstiden gånger skillnaden mellan ärvärdet och börvärdet.

$$Td * (\frac{d}{dt} * (r(t) - y(t))) \quad (5)$$



Figur 5. Flödesschema för en PID-regulator.

Beroende på hur parametrarna konfigureras kan processens beteende ändras. En högre proportionell förstärkning, P-värde, ger en snabbare respons men med sämre stabilitet med risk för oscillerande översvängningar vid för höga värden. En lägre integrationstid, I-värde, ger bättre kompensering för konstanta och lågfrekventa störningar men sämre stabilitet. En högre deriveringstid, D-värde, ger bättre stabilitet genom att förhindra översvängningar men större påverkning av mätfel och signalbrus. För hög deriveringstid kan orsaka instabilitet [6].

3.4 HMI

En HMI-skärm är en skärm som används inom industrin för att övervaka eller styra processer. En HMI-skärm används ofta för att öka användbarheten av PLC-system, i stället för att använda fysiska knappar och läsa av givare manuellt kan ett användargränssnitt designas för en HMI-skärm som gör att man kan styra processen via skärmen och läsa av alla sensorer från samma ställe.

Användargränssnittet designas av en programmerare och överförs till HMI-skärmen, varefter skärmen kan anslutas direkt till en PLC och användas utan externa enheter. Gränssnittet finns lagrat i skärmens lagringsminne.

En HMI-skärm kan variera i storlek och ha olika funktioner som bland annat knappar, pekskärm och trådlös uppkoppling [7].

3.5 Liknande system

Liknande projekt har tidigare gjorts för Tallinn University of Technology där man styrde ett flertankssystem via en Raspberry Pi [8]. I det projektet användes en Raspberry Pi för styrningen och programmeringen skedde i programmeringsspråket Python.

Detta system kan ses som ett parallellprojekt eftersom målet för de båda projekten var det samma, att skapa ett alternativ till MATLAB-programmet som fanns tillgängligt, men tillvägagångssätten var helt olika.

Tidigare projekt som bidrog till mitt examensarbete var ett praktikprojekt för Yrkeshögskolan Novia där interface elektroniken för flertankssystemet designades av en studerande [9]. Denna interface läste av nivågivarna i flertankssystemet och förstärkte signalen från dessa till 0–10 VDC och den förstärkta signalen anslöts till en PLC som jag använde i mitt projekt.

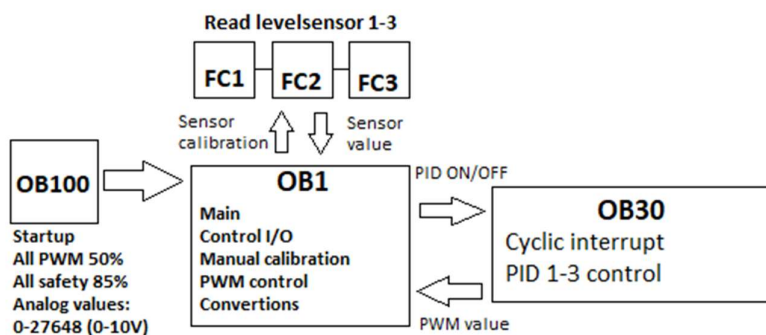
4 Metoder

Projektet inleddes med att testa MATLAB-programmet och se vilka funktioner som skulle återskapas i PLC-programmet. De viktigaste funktionerna som skulle återskapas var manuell styrning av enskilda komponenter, PWM-styrning, kalibreringsmöjligheter, testknappar för flertankssystemet (Figur 6) samt PID-styrning.



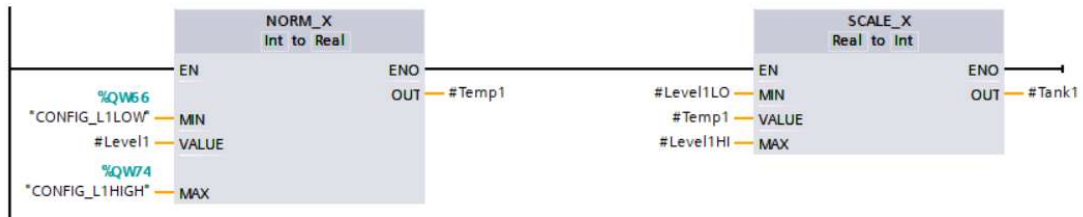
Figur 6. Testknapparna från MATLAB-programmet.

Programmeringen utfördes i TIA-portal v16 och testades via Siemens PLCsim och senare på plats i Technobothnia.

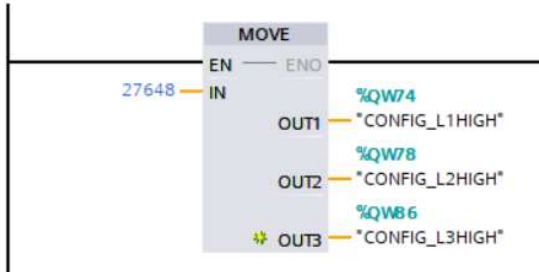


Figur 7. Blockdiagram över PLC-strukturen.

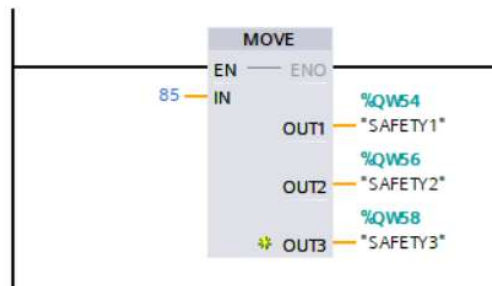
Först skapades ett funktionsblock som läste av värdet från nivågivarna (Figur 8) och konverterade detta till en signal på 0–100, därefter skapades ett nödstopp (Figur 11) som skulle stänga av pumpen/ventilerna ifall vattennivån blev för hög. Nivågivarnas kalibreringsvärden från användargränssnittet bildar funktionens absoluta minimi- och maximivärden som används vid konverteringen 0–100. Vid systemets uppstartning återgår kalibreringsvärdena till 0 och 27 648 (Figur 9), som motsvarar 0 och 10 V. Maxnivån är på 85 % när programmet startas (Figur 10) men kan konfigureras till önskad nivå via HMI-gränssnittet.



Figur 8. Funktion för avläsning av nivågivare.



Figur 9. Startup-konfiguration av nivågivarnas maxvärden.

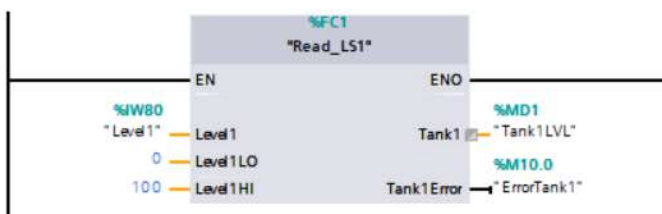


Figur 10. Startup-konfiguration av maxnivåer för tankarnas uppfyllnad.



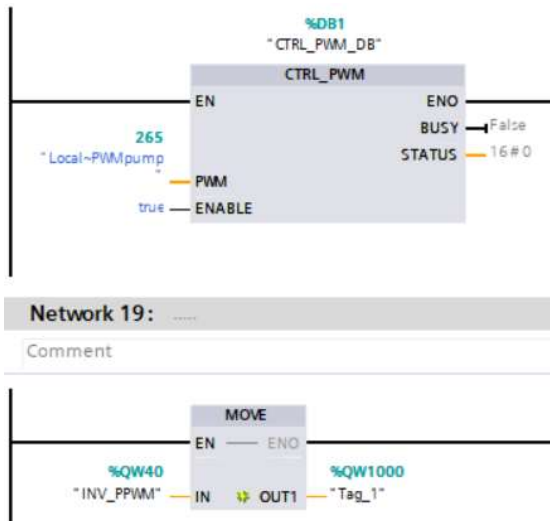
Figur 11. Nödstopp vid angiven maxnivå.

Nivågivarna läses av i början av main-loopen eftersom de borde uppdateras så ofta som möjligt och inte är beroende av att konstant köras med ett jämnt intervall (Figur 12).



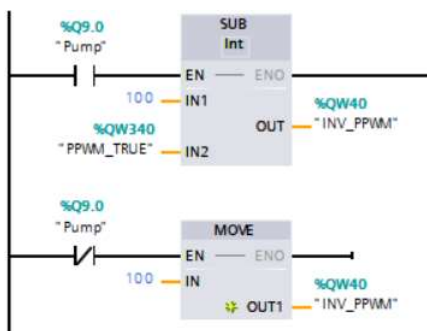
Figur 12. Avläsning av nivågivare.

PWM-blocken (Figur 13) körs kontinuerligt i bakgrunden i main-loopen och när komponenterna stängs av skrivs värdet 100 till pulsbreddsmoduleringen. Eftersom styrningen är inverterad motsvarar styrsignalen 100 avstängning. Inledningsvis stängdes PWM-blocken helt och hållet av vid avstängning men detta ändrades eftersom det ledde till oönskat beteende.



Figur 13. PWM-blocken.

Eftersom styrsignalerna till pulsbreddsmoduleringen behövde inverteras användes ett SUBTRACT-block för att subtrahera önskat styrvärde från 100 (Figur 14). Detta sker i main-loopen.

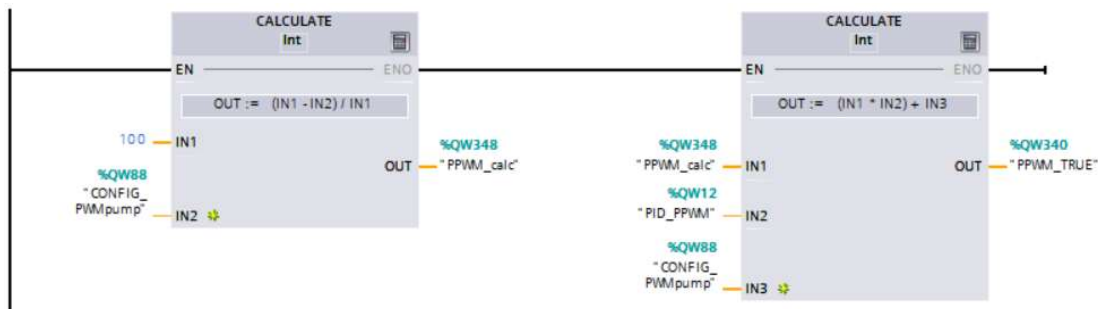


Figur 14. Invertering av pulsbreddsmoduleringens styrsignal.

För att kalibrera pulsbreddsmoduleringen användes den matematiska formeln:

$$\frac{100-Cal}{100} * PWM + Cal \quad (6)$$

Cal=kalibreringsvärde, PWM=pulsbreddsmoduleringens styrvärde.



Figur 15. Pulsbreddsmoduleringens kalibreringsformel.

Denna formel (Figur 15) gör att man kan ange ett kalibreringsvärde som blir den nya nollpunkten för pulsbreddsmoduleringen i programmet och resten av styrningen fortsätter från 0–100 %.

Tabell 1. Pulsbreddsmoduleringens signaler med olika kalibreringsvärden.

PWM%	Okalibrerat	Cal 10%	Cal 25%
0	0	10	25
10	10	19	32.5
20	20	28	40
30	30	37	47.5
40	40	46	55
50	50	55	62.5
60	60	64	70
70	70	73	77.5
80	80	82	85
90	90	91	92.5
100	100	100	100

Analoga ingångarna kan också kalibreras från HMI-interfacen. Detta görs genom att de värden som används för att konvertera analoga värdet till 0–100 är justerbara från HMI-interfacen. Ursprungsvärdena är 0–27 648, men man kan justera den nedre och övre gränsen enligt behov. Efter justeringen används de nya värdena som minimi- och maximivärden som motsvarar 0 och 100 %.

Standardvärdena återgår till 0 och 27 648 när man startar om systemet, dessa värden ger 0 V -> 0 och 10 V -> 27 648 som sedan konverteras till 0–100 %. Alla kalibreringsparametrar justeras från en skild kalibreringssida i HMI-interfacen (se figur 18).

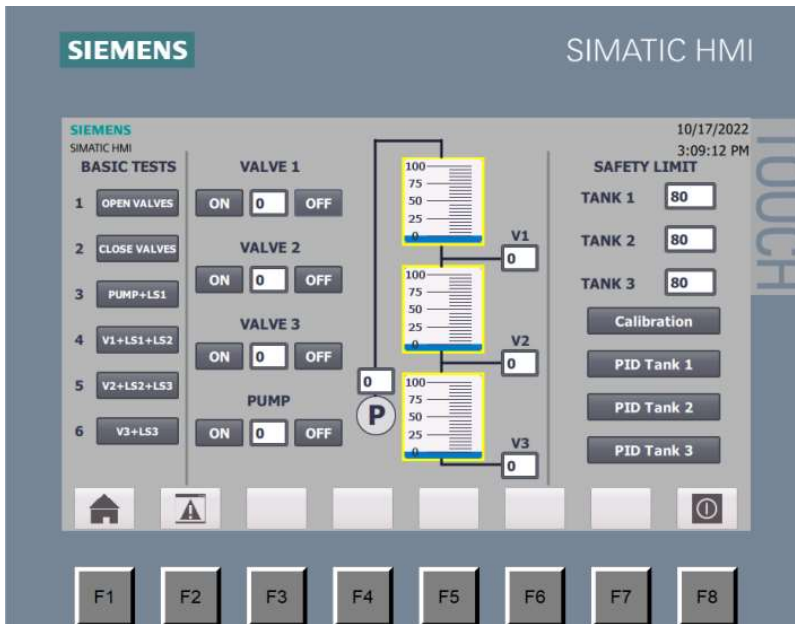
Manuella styrningen sker genom att "ON" knapparna triggat en SET-funktion och "OFF" knapparna triggat en RESET-funktion (Figur 16). När SET-funktionen är på skrivs det värde som anges i HMI-interfacen till PWM-styrningen och när SET-funktionen är av skrivs 100 till PWM-styrningen.



Figur 16. Manuell styrknapp.

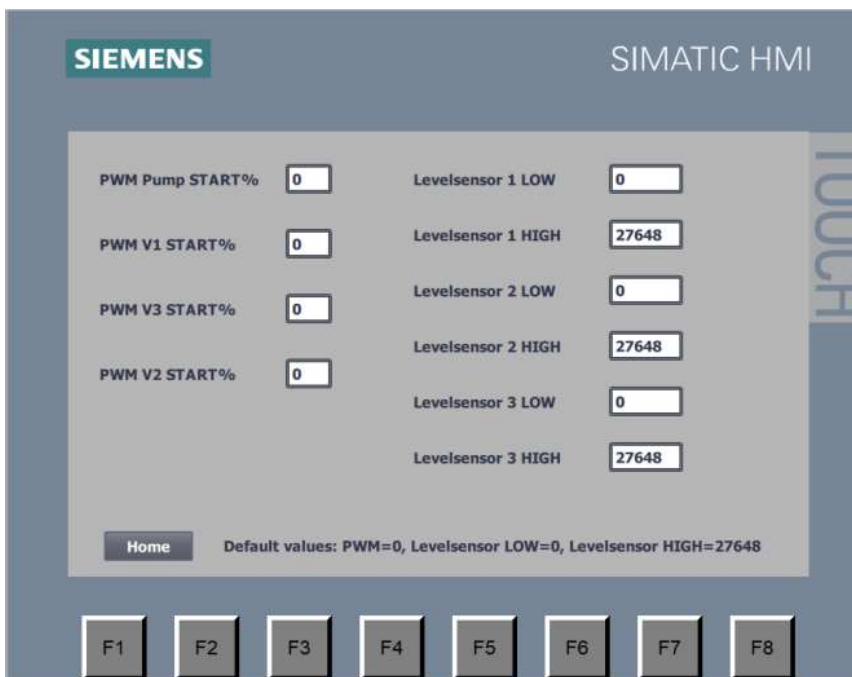
Ett HMI-interface designades som skulle användas med en SIMATIC KTP700 Basic skärm.

På första sidan av HMI-interfacen ser man alla testknappar, en visuell överblick av vattennivåerna i tankarna, manuella styrknappar samt knappar som tar användaren till resten av sidorna (Figur 17).



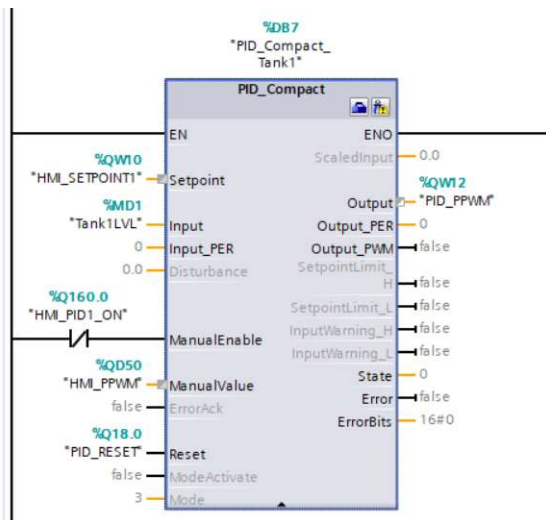
Figur 17. HMI-interface, hemskärm.

På de övriga sidorna finns kalibrering av alla ingångar och utgångar (Figur 18) samt en egen sida för varje PID-styrning (Figur 21).

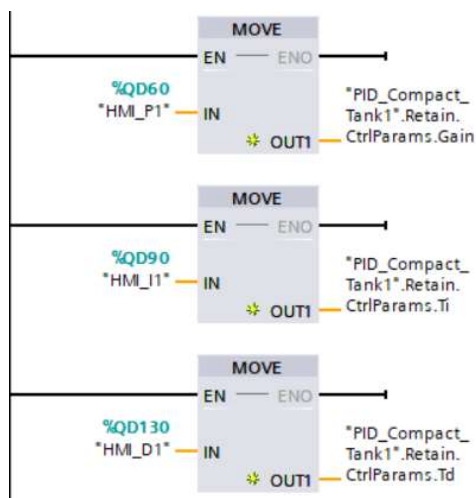


Figur 18. HMI-interface, kalibreringsskärm.

PID-funktionerna sker kontinuerligt i bakgrunden med en 200 ms cyclic-interrupt. Eftersom flertanksprocessen är relativt långsam ansågs 200 millisekunder vara en tillräcklig samplingstid, för snabbare processer kunde tiden justeras till 100, 60 eller 80 ms. När PID-styrningen är avaktiverad ger den ut samma signal som den manuella PWM-styrningen på startsidan, man kan alltså styra de komponenterna som inte används i den aktiva PID-styrningen manuellt från startskärmen. När PID-styrningen är aktiverad styr den automatiskt komponenterna mot angiven setpoint enligt de P-, I- och D-värden som användaren har angett på skärmen. Om man vill köra en PI- eller PD-styrning lämnar man endast den önskade parametern som 0 i interfacen vilket PID-Compact funktionen (Figur 19) tolkar som att parametern ej används. P-parametern anges som förstärkning medan I- och D-parametrarna anges som tid (Figur 20).



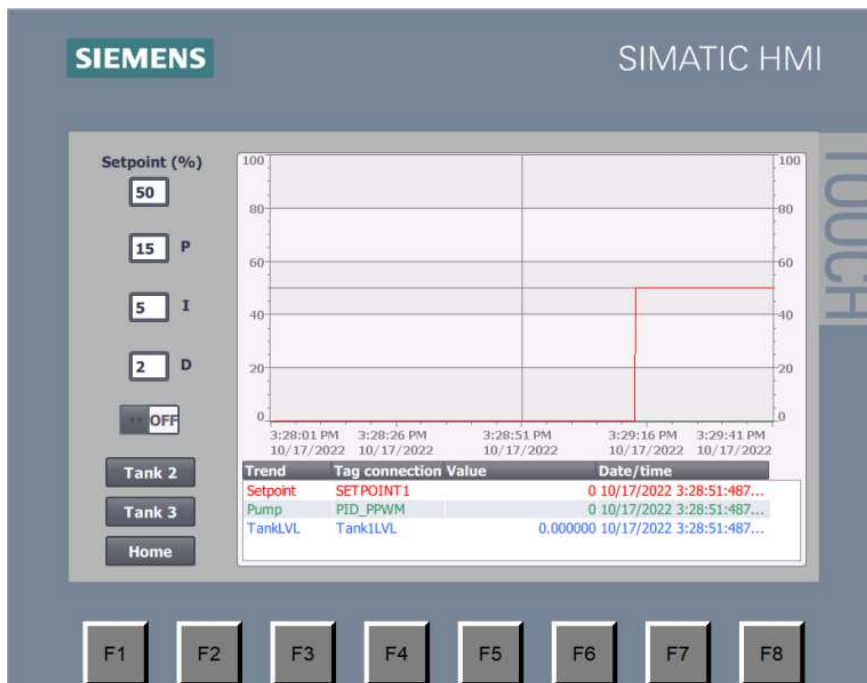
Figur 19. PID-styrningens funktionsblock.



Figur 20. PID-styrningens P-, I- och D-parametrar.

På PID sidan ser man även en graf som loggar värdet på olika komponenter. För tank 1 ser man pumpens aktivitet, vattennivån i tank 1 samt den angivna setpointen. För tank 2 ser man aktiviteten för ventil 1, vattennivån i tank 2 samt den angivna setpointen. För tank 3 ser man aktiviteten för ventil 2, vattennivån i tank 3 samt den angivna setpointen. Dessa värden loggas med 999 punkter så man får en längre överblick av aktiviteten. Skalan på Y-axeln är inställd som 0–100, vilket representerar nivåerna samt komponenterna i en procentskala.

Med PID-styrningen styrs pumpen av PID1, bottenventilen i översta tanken av PID2 och bottenventilen i mellersta tanken av PID3. Nedersta tankens bottenventil kan öppnas manuellt via HMI-skärmen eller med hjälp av vridventilen på flertankssystemet.



Figur 21. HMI-interface, PID-kontrollskärm.

5 Resultat

Projektets PLC-program är färdigt och det innehåller de funktioner som ansågs vara väsentliga. HMI-interfacet är intuitivt och modernare än det ursprungliga MATLAB-interfacet.

Förutom examensarbetet skrevs en handbok som kan användas av framtida användare för att underlätta användningen av HMI-interfacet. Med justeringar kan man ändra programmet om man vill. Några exempel på justeringar är att ändra vattennivån till liter i stället för procent genom att lägga till uträkningsfunktioner som konverterar värdet från nivågivarna till volymen beroende på tankens utformning eller att styra PWM med ‰, 0–1000, i stället för %, 0–100. Jag hoppas att fler studeranden kan expandera på vad som redan finns tillgängligt för att vidareutveckla programmet i framtiden.

6 Diskussion

I framtiden skulle några funktioner kunna läggas till i programmet, till exempel kaskadstyrning och open-loopstyrning, och vissa justeringar skulle kunna göras i det befintliga programmet.

Största problemet med programmet i nuläget är att mätvärdet från analoga ingångarna ibland sjunker till 0 momentärt. Denna momentära minskning beror troligtvis på att PID-styrningen sker via en cyclic-interrupt som avbryter main-loopen tillfälligt, vilket gör att mätvärden sjunker till 0.

Ett sätt att fixa detta skulle vara att köra PID-styrningen i main-loopen men detta rekommenderas ej eftersom PID-styrningen blir mer exakt om den får köras separat. Ett annat sätt vore att logga värdet från sensorerna och återge det senaste mätvärdet ifall de under en kort stund tappar signalen.

7 Sammanfattning

Styrningen av ett flertankssystem av tillverkaren Inteco programmerades till en Siemens S7 1215C DC/DC/DC PLC. Ett användargränssnitt skapades för en tillhörande SIMATIC KTP700 Basic HMI-skärm. Från HMI-skärmen kan man kalibrera nivågivare och PWM-utgångar, styra komponenter enskilt med PWM-styrning, välja maxnivå för vattennivån i flertankssystemet, testa flertankssystemets komponenter samt köra PID-styrning på en eller flera tankar.

7.1 Skribentens egna anmärkningar

Jag hoppas att detta projekt kommer att användas i många kommande år och att flertankssystemet kommer att förbli en del av utbildningen inom el- och automationslinjen på Yrkeshögskolan Novia. Projektet har varit lärorikt och intressant och jag hoppas att framtida studeranden kommer att ta del av det jag har skapat och möjligtvis expandera på det i framtiden. Jag vill också rikta ett stort tack till Joachim Böling som har varit väldigt hjälpsam genom att ge feedback och konstruktiv kritik samt komma med önskemål på funktioner och ändringar i interfacet. Utan hans kommentarer skulle många funktioner troligtvis saknas och slutresultatet skulle inte ha varit lika användbart för framtida bruk.

8 Källor

- [1] Inteco Ltd., Multitank system user's manual, Krakow.
- [2] Siemens, S7-1200 Programmable controller System Manual, München, 2012.
- [3] T. M. Antonsen, PLC Controls with Ladder Diagram (LD), Köpenhamn: Books on Demand GmbH, 2021.
- [4] B. G. Lipták, Process Control and Optimization, Oxfordshire: Taylor & Francis, 2018.
- [5] M. Barr, "Pulse Width Modulation," *Embedded Systems Programming*, pp. 103-104, 2001.
- [6] K. J. Åström, PID Controllers: Theory, Design, and Tuning, North Carolina: ISA, 1995.
- [7] H. Berger, Automating with SIMATIC, Erlangen: Publicis Publishing, 2016.
- [8] A. Smolov, "Control of multitank system using Raspberry Pi," Tallinn University of Technology, Department of Computer Systems, Tallinn, 2017.
- [9] "Teknisk dokumentering av kretskort till multitank simulator," Yrkeshögskolan Novia, Vasa, 2022.